**PROG8170**

**Software Quality Assurance Techniques**

**Spring 2023**

**Midterm Exam Submission**

Submitted by:

|  |  |  |
| --- | --- | --- |
| First Name | Last Name | Student ID |
| Siyu | Liu | 8859412 |

June 2023

**Question #1 Submission**



**Question #2 Submission**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Analyze: The application is not ready to be released. Firstly, the page for converting "Volume and Capacity" is not available, this will lead to considerable inconvenience for users. Secondly, the unit converting results are obviously incorrect, this will make the application unreliable. Thirdly, some defects are not as serious as the above ones, like the return menu link is disabled on “Covert Temperature” page, but these bugs should be fixed properly before the application is released.

**Question #3 Submission**

{Provide your response to Question #3 here}

A screenshot of a computer

Description automatically generated

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using NUnit.Framework;

using PROG8170Midterm;

namespace TestTemp

{

[TestFixture]

internal class Program

{

static void Main(string[] args)

{

}

[Test]

public void Temp\_Input0f\_ExpectCold()

{

// Arrage

float inputTemp = 0;

string expectOutcome = "Brrr...it is cold.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input5f\_ExpectCold()

{

// Arrage

float inputTemp = 5;

string expectOutcome = "Brrr...it is cold.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input9dot9f\_ExpectCold()

{

// Arrage

float inputTemp = 9.9f;

string expectOutcome = "Brrr...it is cold.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input10f\_ExpectNice()

{

// Arrage

float inputTemp = 10;

string expectOutcome = "It is nice outside.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input15f\_ExpectNice()

{

// Arrage

float inputTemp = 15;

string expectOutcome = "It is nice outside.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input19dot9f\_ExpectNice()

{

// Arrage

float inputTemp = 19.9f;

string expectOutcome = "It is nice outside.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input20f\_ExpectComfortable()

{

// Arrage

float inputTemp = 20;

string expectOutcome = "It is very comfortable.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input25f\_ExpectComfortable()

{

// Arrage

float inputTemp = 25;

string expectOutcome = "It is very comfortable.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input29dot9f\_ExpectComfortable()

{

// Arrage

float inputTemp = 29.9f;

string expectOutcome = "It is very comfortable.";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input30f\_ExpectHot()

{

// Arrage

float inputTemp = 30;

string expectOutcome = "It is boling hot outside!";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input35f\_ExpectHot()

{

// Arrage

float inputTemp = 35;

string expectOutcome = "It is boling hot outside!";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input40f\_ExpectHot()

{

// Arrage

float inputTemp = 40;

string expectOutcome = "It is boling hot outside!";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_InputMinus1f\_ExpectInvalid()

{

// Arrage

float inputTemp = -1;

string expectOutcome = "Invalid Temperature";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input41f\_ExpectInvalid()

{

// Arrage

float inputTemp = 41;

string expectOutcome = "Invalid Temperature";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

[Test]

public void Temp\_Input45dot5f\_ExpectInvalid()

{

// Arrage

float inputTemp = 45.5f;

string expectOutcome = "Invalid Temperature";

// Act

string actualOutcome = Temp.TemperatureConverter(inputTemp);

// Assert

Assert.AreEqual(expectOutcome, actualOutcome);

}

}

}